

REPORT

ZDEMO_INTERCTIVE_RPRTS_BUTTONS.

types : begin of ty_lfa1 , " *from vender master table(general section)*

lifnr type lifnr , " Account Number of Vendor or Creditor
name1 type name1_gp, " name
ort01 type ort01_gp, " city
land1 type land1_gp, " country key
end of ty_lfa1.

types : begin of ty_ekko , " Purchasing Document Header
ebeln type ebeln , "Purchasing Document Number
aedat type erdat , "Date on Which Record Was Created
LIFNR TYPE LIFNR,
end of ty_ekko.

data : it_lfa1 type table of ty_lfa1,
 it_ekko type table of ty_ekko,
 wa_lfa1 type ty_lfa1,
 wa_ekko type ty_ekko,
 v_lifnr type lifnr,
 V_NAME1 TYPE NAME1_GP,
 IT_SELECTED_LFA1 TYPE TABLE OF TY_LFA1,
 v_box(1) type c.

select-options s_lifnr for v_lifnr default 1 to 100.

select lifnr name1 ort01 land1 from lfa1 into table it_lfa1 where
lifnr in s_lifnr.

```
SET PF-STATUS 'ZSTATUS'.  
perform display_data.
```

```
sy-lsind = 0.
```

```
AT USER-COMMAND.  
CASE SY-UCOMM.
```

```
WHEN 'SEAL'.  
V_BOX = 'X' . " CHECKED.  
PERFORM DISPLAY_DATA.  
WHEN 'DSEAL'.  
V_BOX = ' '. "DESELECT  
PERFORM DISPLAY_DATA.  
WHEN 'DOWN'.  
REFRESH IT_SELECTED_LFA1.  
DATA V_LINES TYPE i.
```

```
* V_LINES = LINES( IT_SELECTED_LFA1 ).
```

```
DESCRIBE LIST NUMBER OF LINES V_LINES.
```

```
* WRITE : / 'V_LINES ', V_LINES.
```

```
DO V_LINES TIMES.
```

```
CLEAR WA_LFA1.  
READ LINE SY-INDEX FIELD VALUE V_BOX WA_LFA1-LIFNR.
```

```
IF V_BOX = 'X'.  
CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'  
EXPORTING  
  INPUT      = WA_LFA1-LIFNR  
IMPORTING  
  OUTPUT     = WA_LFA1-LIFNR  
.
```

```
    READ TABLE IT_LFA1 INTO WA_LFA1 WITH KEY lifnr = WA_LFA1-  
LIFNR.
```

```
    IF SY-SUBRC = 0.  
        APPEND WA_LFA1 TO IT_SELECTED_LFA1.  
    ENDIF.  
    CLEAR : WA_LFA1, V_BOX.  
    ENDIF.  
    ENDDO.
```

```
    IF IT_SELECTED_LFA1 IS NOT INITIAL.  
        CALL METHOD  
CL_GUI_FRONTEND_SERVICES=>GUI_DOWNLOAD  
EXPORTING  
    FILENAME = 'D:/BUTTON/BUTTON.TXT'  
    write_field_separator = ''  
    CHANGING  
        DATA_TAB = IT_SELECTED_LFA1.
```

```
    IF SY-SUBRC = 0.  
        MESSAGE 'SUCCESSFUL DOWNLOADED' TYPE 'S'. " STATUS  
MESSAGE  
    ENDIF.
```

```
    ELSE.  
        MESSAGE 'NO DATA TO DOWNLOAD ' TYPE 'S'. " STATUS  
MESSAGE  
    ENDIF.
```

```
    .
```

```
    WHEN 'GETPOS'.  
        PERFORM COLLECT_SELECTED_LINES.
```

```
    IF IT_SELECTED_LFA1 IS NOT INITIAL .
```

```
SELECT EBELN AEDAT LIFNR INTO TABLE IT_EKKO FROM EKKO
FOR ALL ENTRIES IN IT_SELECTED_LFA1
WHERE LIFNR = IT_SELECTED_LFA1-LIFNR.
```

```
IF IT_EKKO IS NOT INITIAL.
SORT IT_EKKO BY LIFNR EBELN.
LOOP AT IT_EKKO INTO WA_EKKO.
WRITE :/ SY-TABIX, WA_EKKO-EBELN, WA_EKKO-LIFNR,
WA_EKKO-AEDAT.
ENDLOOP.
ELSE.
WRITE :/ 'NO PURCHASE FOUND FROM VENDER'.
```

```
ENDIF.
```

```
ELSE.
MESSAGE 'NO VENDER SELECTED FROM THE LIST ' TYPE 'S'.
ENDIF.
```

```
ENDCASE.
```

```
FORM DISPLAY_DATA.
LOOP AT it_lfa1 into wa_lfa1.
write :/ V_BOX AS CHECKBOX , wa_lfa1-lifnr color 1 , wa_lfa1-
name1 color 3 , wa_lfa1-ort01 color 5, wa_lfa1-land1 color 7.
ENDLOOP.
ENDFORM.
```

```
FORM COLLECT_SELECTED_LINES .
REFRESH IT_SELECTED_LFA1.
DATA V_LINES TYPE I.
```

DESCRIBE LIST NUMBER OF LINES V_LINES.

DO V_LINES TIMES.

CLEAR WA_LFA1.

READ LINE SY-INDEX FIELD VALUE V_BOX WA_LFA1-LIFNR.

IF V_BOX = 'X'.

CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'

EXPORTING

INPUT = WA_LFA1-LIFNR

IMPORTING

OUTPUT = WA_LFA1-LIFNR

.

READ TABLE IT_LFA1 INTO WA_LFA1 WITH KEY LIFNR =
WA_LFA1-LIFNR.

IF SY-SUBRC = 0 .

APPEND WA_LFA1 TO IT_SELECTED_LFA1.

ENDIF.

CLEAR : WA_LFA1, V_BOX.

ENDIF.

ENDDO.

ENDFORM.